




PROCESAMIENTO Y LIMPIEZA DE
DATOS

LIMPIEZA DE UNA BASE DE DATOS ENSUCIADA

Pavel Alarcon Miranda
Miercoles y Viernes

RECEPCION DE LA BASE DE DATOS E IMPORTACION DE LIBRERIAS

```
#Importo la base de datos
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
url = 'df_sucio.csv'
df=pd.read_csv(url)
df.head(5)
```

[251] ✓ 0.0s  Open 'df' in Data Wrangler

Python

	Hours_Studied	Attendance	Parental_Involvement	Access_to_Resources	Extracurricular_Activities	Sleep_Hours	Previous_Score
0	23.0	84.0	Low	High	No	7.0	
1	19.0	64.0	Low	Medium	No	8.0	
2	24.0	98.0	Medium	Medium	Yes	7.0	
3	29.0	89.0	Low	Medium	NaN	8.0	
4	19.0	NaN	Medium	Medium	Yes	6.0	

ANÁLISIS INICIAL DE LA BASE DE DATOS

Antes de comenzar a limpiar la base de datos, debe realizar un análisis preliminar para comprender la naturaleza y distribución de los errores



ANALISIS DE LA BASE DE DATOS

```
#Tipos de datos
df.info()

✓ 0.0s

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6753 entries, 0 to 6752
Data columns (total 20 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Hours_Studied                        6687 non-null   object
1   Attendance                          6703 non-null   object
2   Parental_Involvement                6695 non-null   object
3   Access_to_Resources                 6696 non-null   object
4   Extracurricular_Activities          6680 non-null   object
5   Sleep_Hours                        6696 non-null   object
6   Previous_Scores                     6687 non-null   object
7   Motivation_Level                    0 non-null      float64
8   Internet_Access                     6700 non-null   object
9   Tutoring_Sessions                  6686 non-null   object
10  Family_Income                       6691 non-null   object
11  Teacher_Quality                     6611 non-null   object
12  School_Type                         6696 non-null   object
13  Peer_Influence                      6687 non-null   object
14  Physical_Activity                   6685 non-null   object
15  Learning_Disabilities               6684 non-null   object
16  Parental_Education_Level            6604 non-null   object
17  Distance_from_Home                  6634 non-null   object
18  Gender                              6686 non-null   object
19  Exam_Score                          6675 non-null   object
dtypes: float64(1), object(19)
memory usage: 1.0+ MB
```

Contiene 6753 filas y 20 columnas

La mayoría de datos son tipo "object"

Todas las columnas tiene algunos valores nulos

REVISION DE DATOS NUMERICOS

```
df['Sleep_Hours'].unique()
```

```
array(['7.0', '8.0', '6.0', '10.0', nan, '9.0', '5.0', 'aaaaa', '4.0'],  
      dtype=object)
```

```
df['Previous_Scores'].unique()
```

```
array(['73.0', '59.0', '91.0', '98.0', '65.0', '89.0', '68.0', '50.0',  
      '80.0', '71.0', '88.0', '87.0', '97.0', '72.0', '74.0', nan,  
      '82.0', '58.0', '99.0', '84.0', '100.0', '75.0', '54.0', '90.0',  
      '94.0', '51.0', '57.0', '66.0', '96.0', '93.0', '56.0', '52.0',  
      '70.0', '63.0', '79.0', '81.0', '69.0', '95.0', '60.0', 'aaaaa',  
      '77.0', '92.0', '62.0', '85.0', '78.0', '64.0', '76.0', '55.0',  
      '86.0', '61.0', '53.0', '83.0', '67.0'], dtype=object)
```

```
df['Motivation_Level'].unique()
```

```
array([nan])
```

```
df['Hours_Studied'].unique()
```

```
array(['23.0', '19.0', '24.0', '29.0', '25.0', '17.0', '21.0', '9.0',  
      '10.0', '14.0', '22.0', '15.0', '12.0', nan, '11.0', '13.0',  
      '16.0', '18.0', '31.0', '20.0', '8.0', '26.0', '28.0', '4.0',  
      'aaaaa', '35.0', '27.0', '33.0', '36.0', '43.0', '34.0', '1.0',  
      '30.0', '7.0', '32.0', '6.0', '38.0', '5.0', '3.0', '2.0', '39.0',  
      '37.0', '44.0'], dtype=object)
```

```
df['Attendance'].unique()
```

```
array(['84.0', '64.0', '98.0', '89.0', nan, '88.0', '78.0', '94.0',  
      '80.0', '97.0', '83.0', '82.0', '68.0', '60.0', '70.0', '75.0',  
      '99.0', '74.0', '65.0', '91.0', '90.0', '66.0', '69.0', '72.0',  
      '63.0', '61.0', '86.0', '77.0', '71.0', 'aaaaa', '67.0', '87.0',  
      '73.0', '96.0', '92.0', '100.0', '81.0', '95.0', '79.0', '76.0',  
      '93.0', '62.0', '85.0'], dtype=object)
```

La columna “Motivation Level” solo contiene datos nulos

REVISION DE DATOS NUMERICOS

```
df['Tutoring_Sessions'].unique()
```

```
array(['0.0', '2.0', '1.0', '3.0', 'aaaaa', '4.0', nan, '5.0', '6.0',  
      '7.0', '8.0'], dtype=object)
```

```
df['Exam_Score'].unique()
```

```
array(['67.0', '61.0', '74.0', '71.0', '70.0', '66.0', '69.0', nan,  
      '68.0', '65.0', '64.0', '60.0', '72.0', '63.0', '62.0', 'aaaaa',  
      '100.0', '76.0', '73.0', '78.0', '89.0', '75.0', '59.0', '86.0',  
      '97.0', '83.0', '84.0', '80.0', '58.0', '94.0', '55.0', '92.0',  
      '77.0', '101.0', '88.0', '79.0', '91.0', '99.0', '87.0', '57.0',  
      '82.0', '96.0', '98.0', '95.0', '85.0', '93.0', '56.0'],  
      dtype=object)
```

Todos los datos son enteros

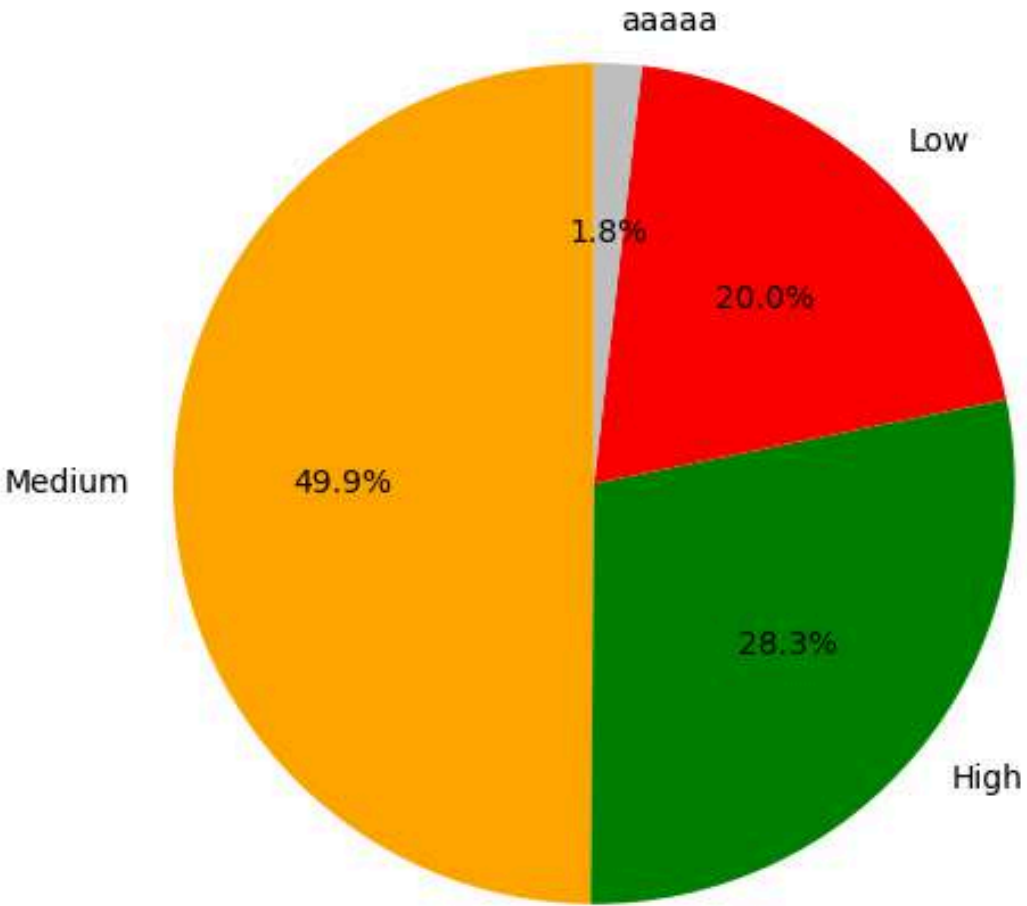
Todas las columnas tienen un valor
invalido llamado "aaaaa"

```
df['Physical_Activity'].unique()
```

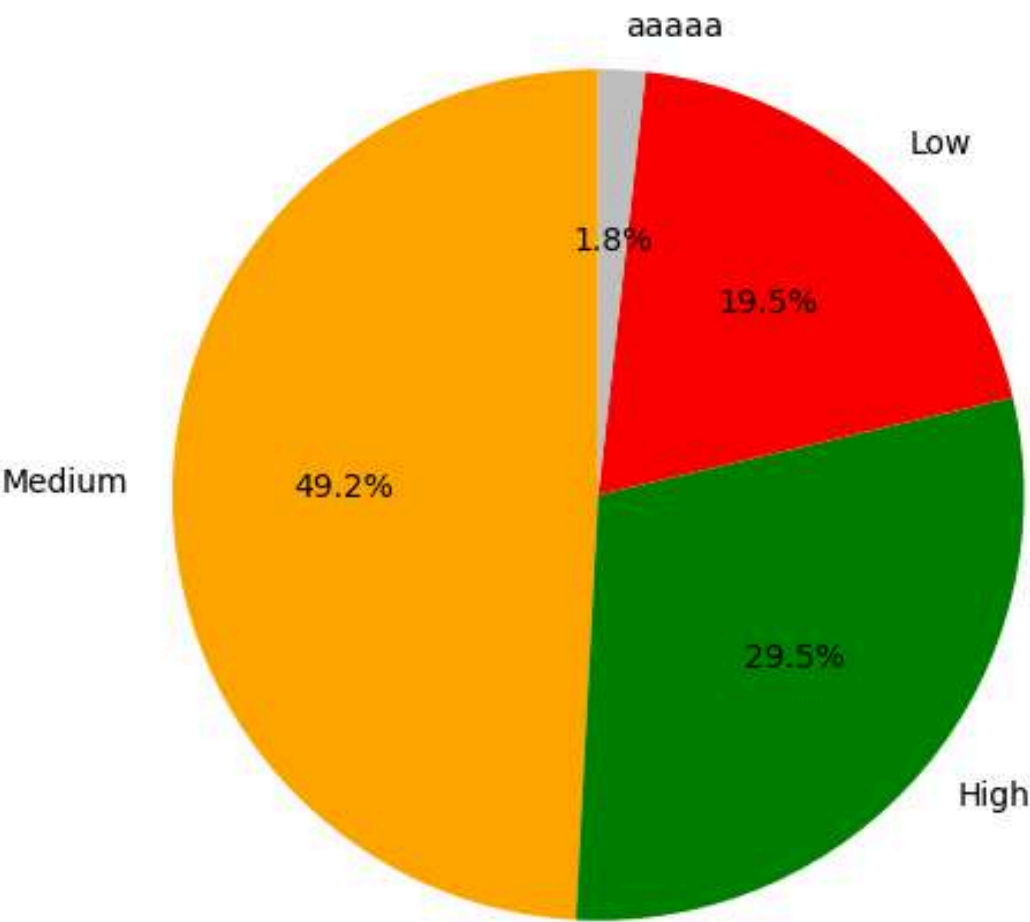
```
array(['3.0', '4.0', '2.0', '1.0', '5.0', nan, 'aaaaa', '0.0', '6.0'],  
      dtype=object)
```


REVISION DE DATOS CATEGORICOS

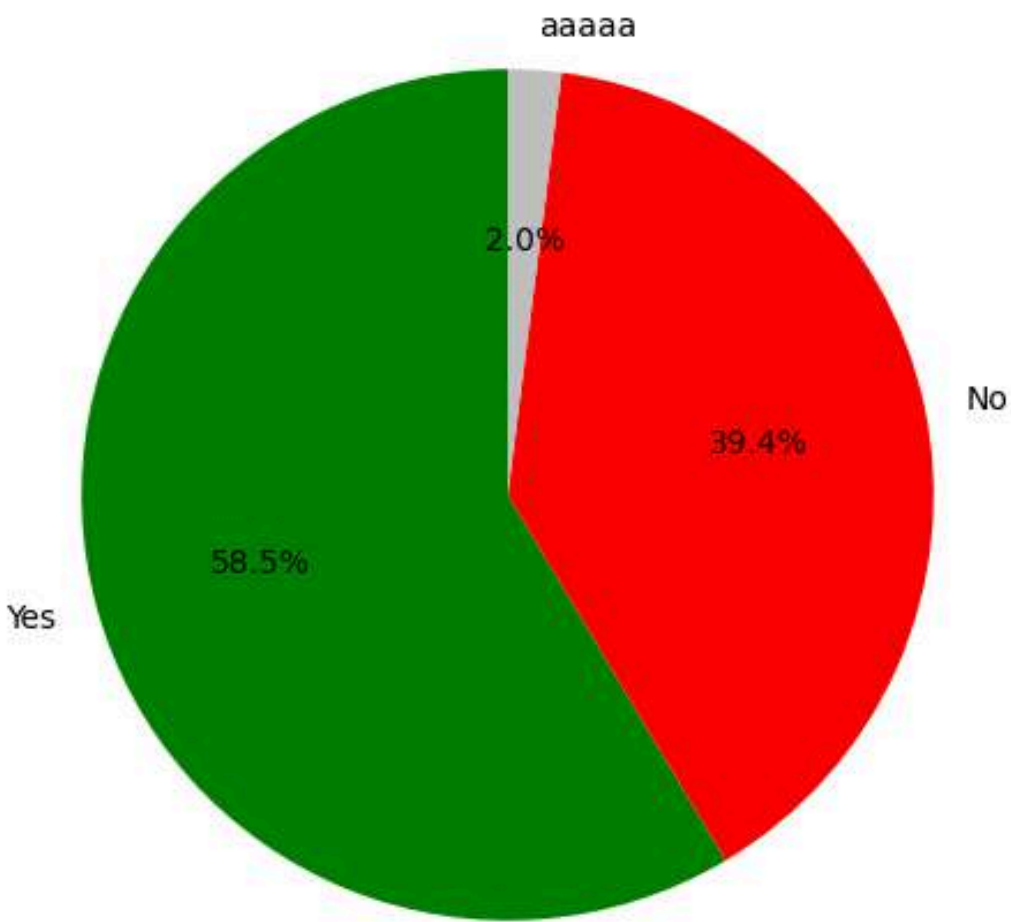
Distribucion por Involucramiento Parental



Distribucion por Acceso a Recursos

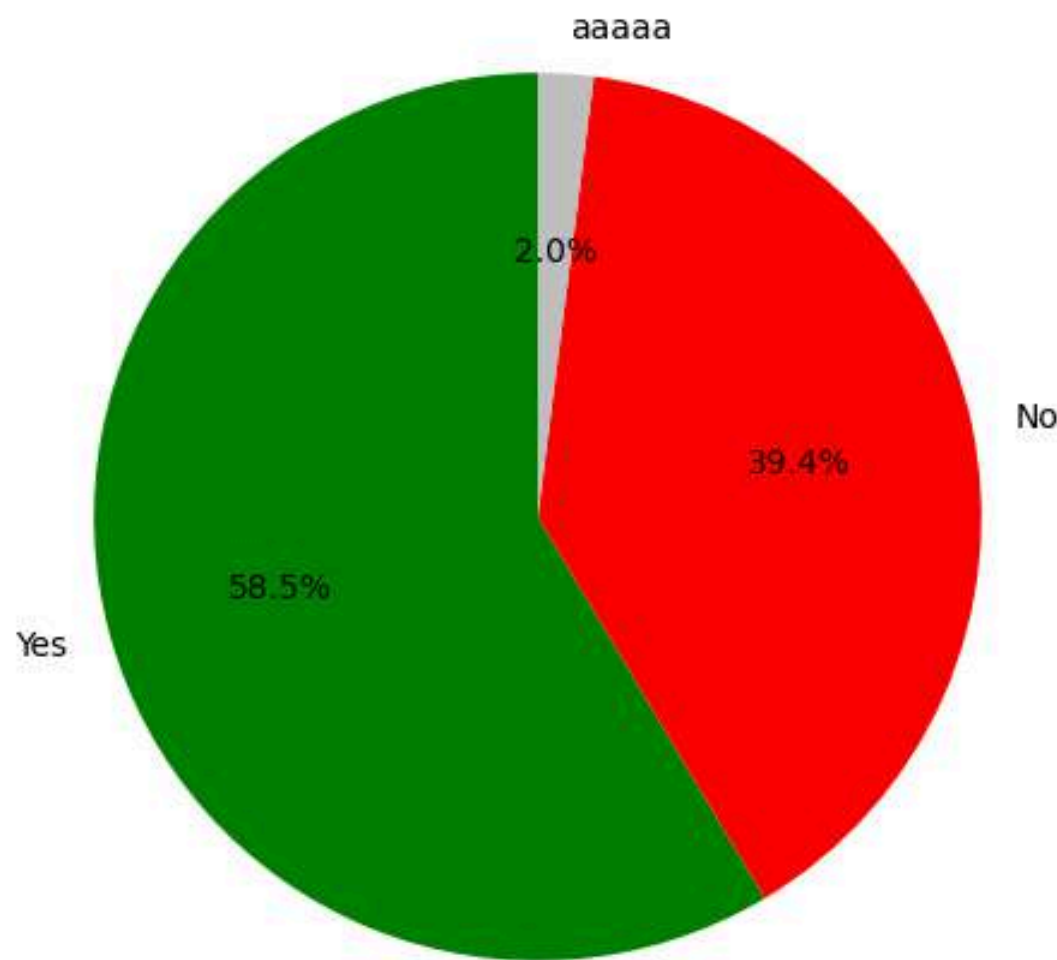


Distribucion por Actividades Extracurriculares

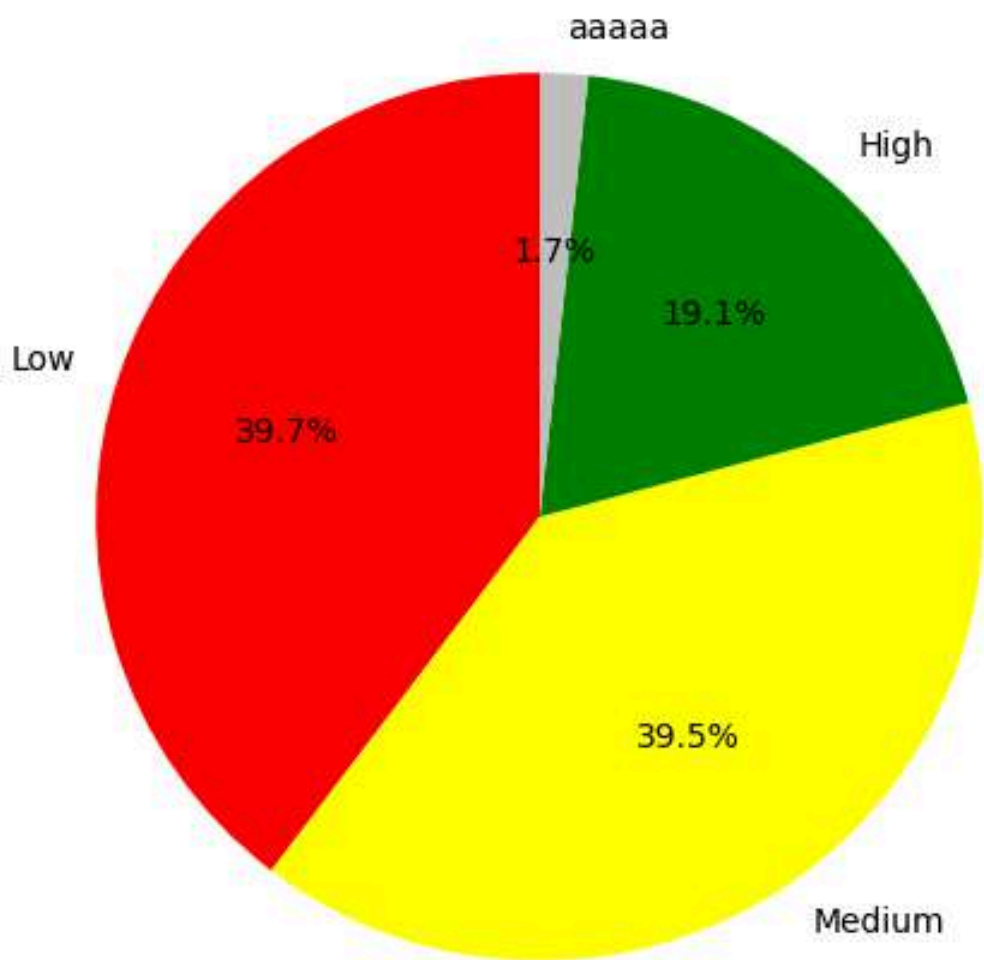


REVISION DE DATOS CATEGORICOS

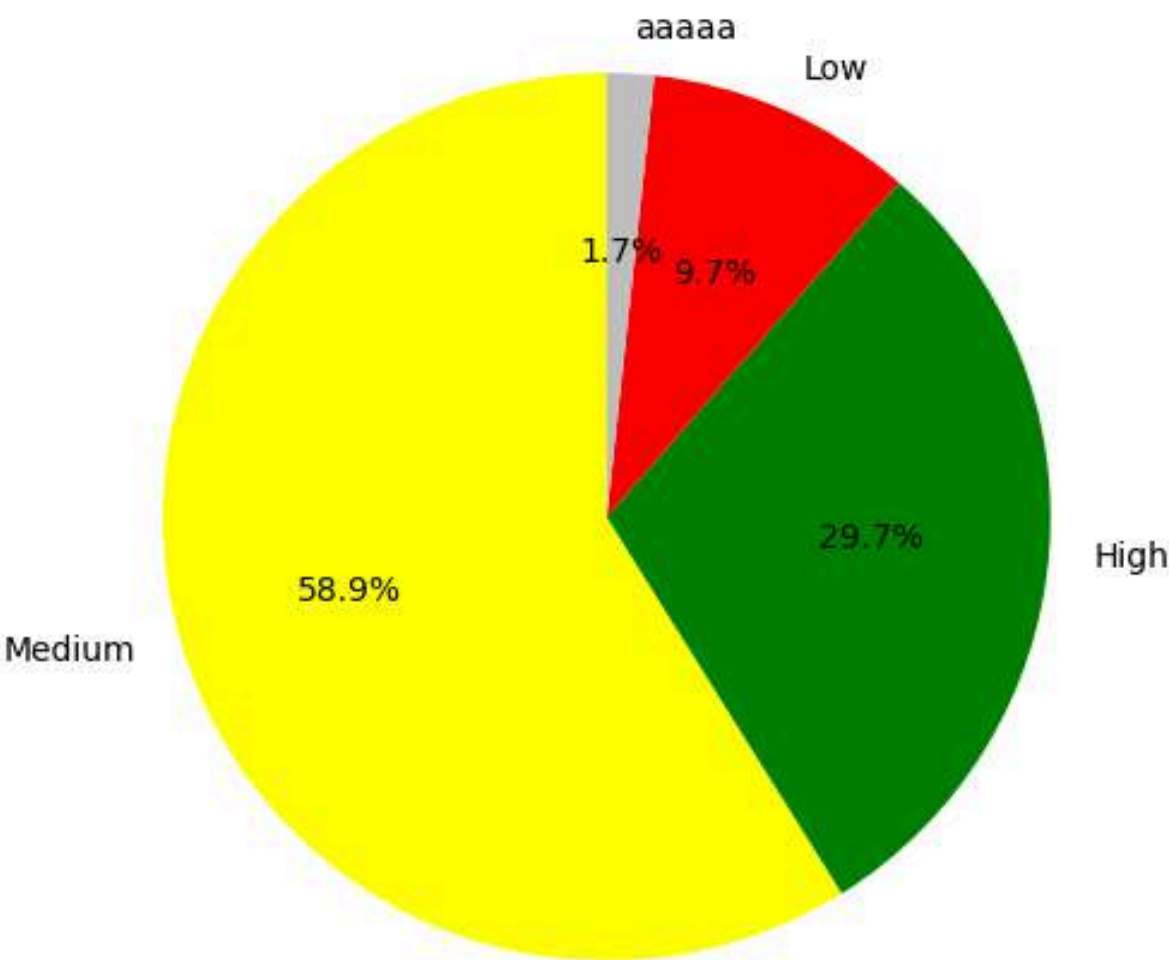
Distribucion por Acceso a Internet



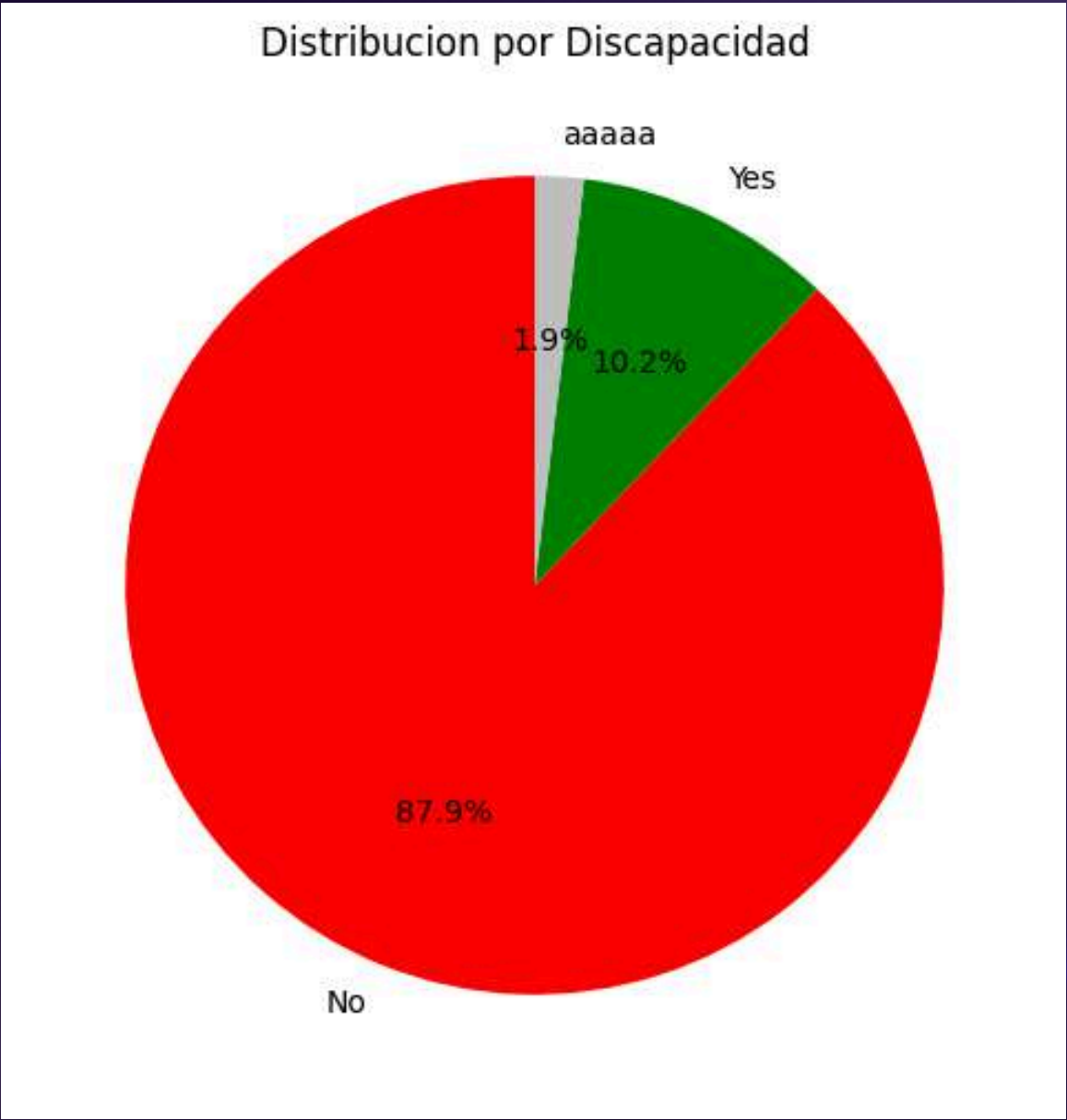
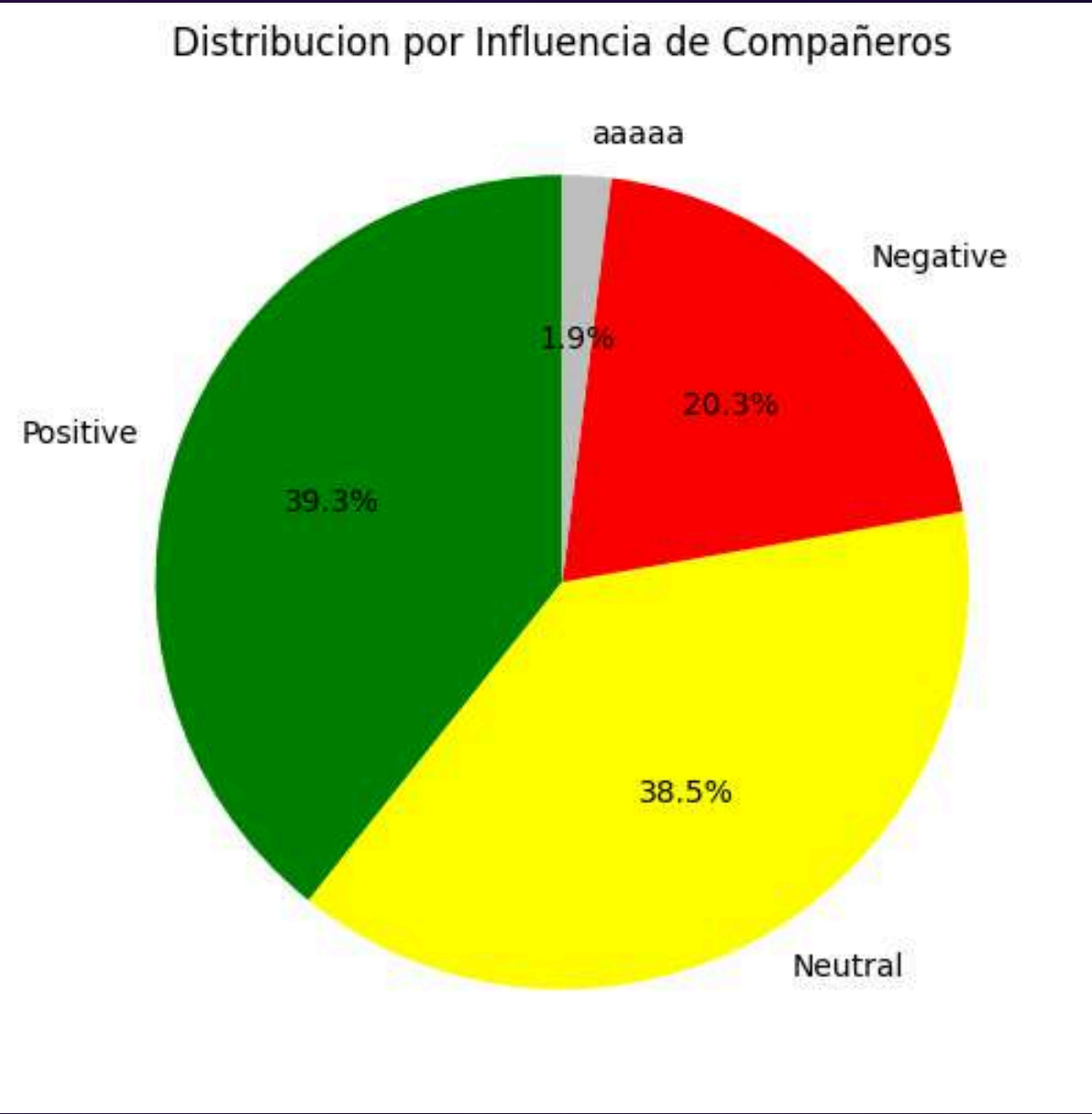
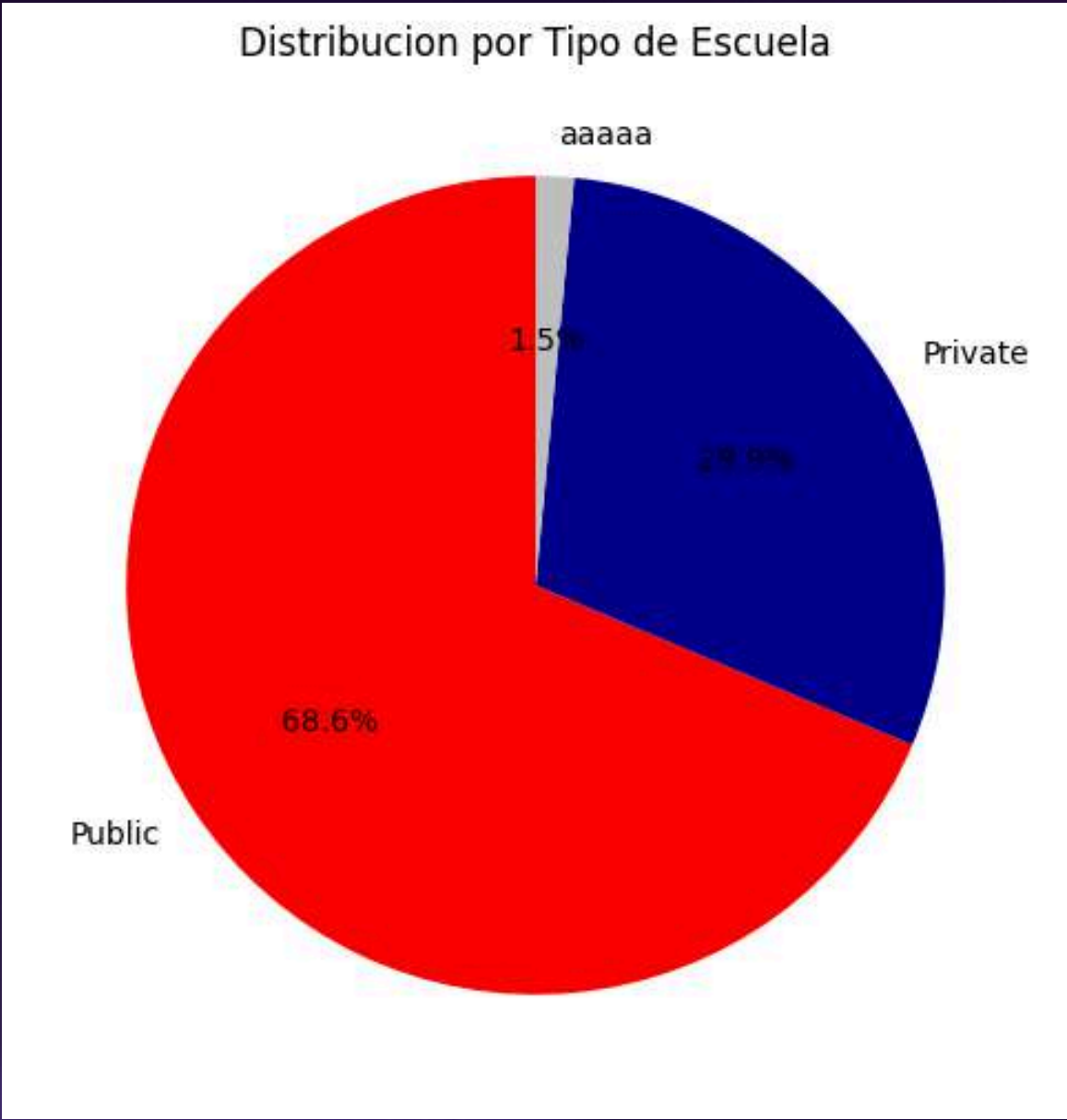
Distribucion por Ingreso



Distribucion por Calidad de Maestro

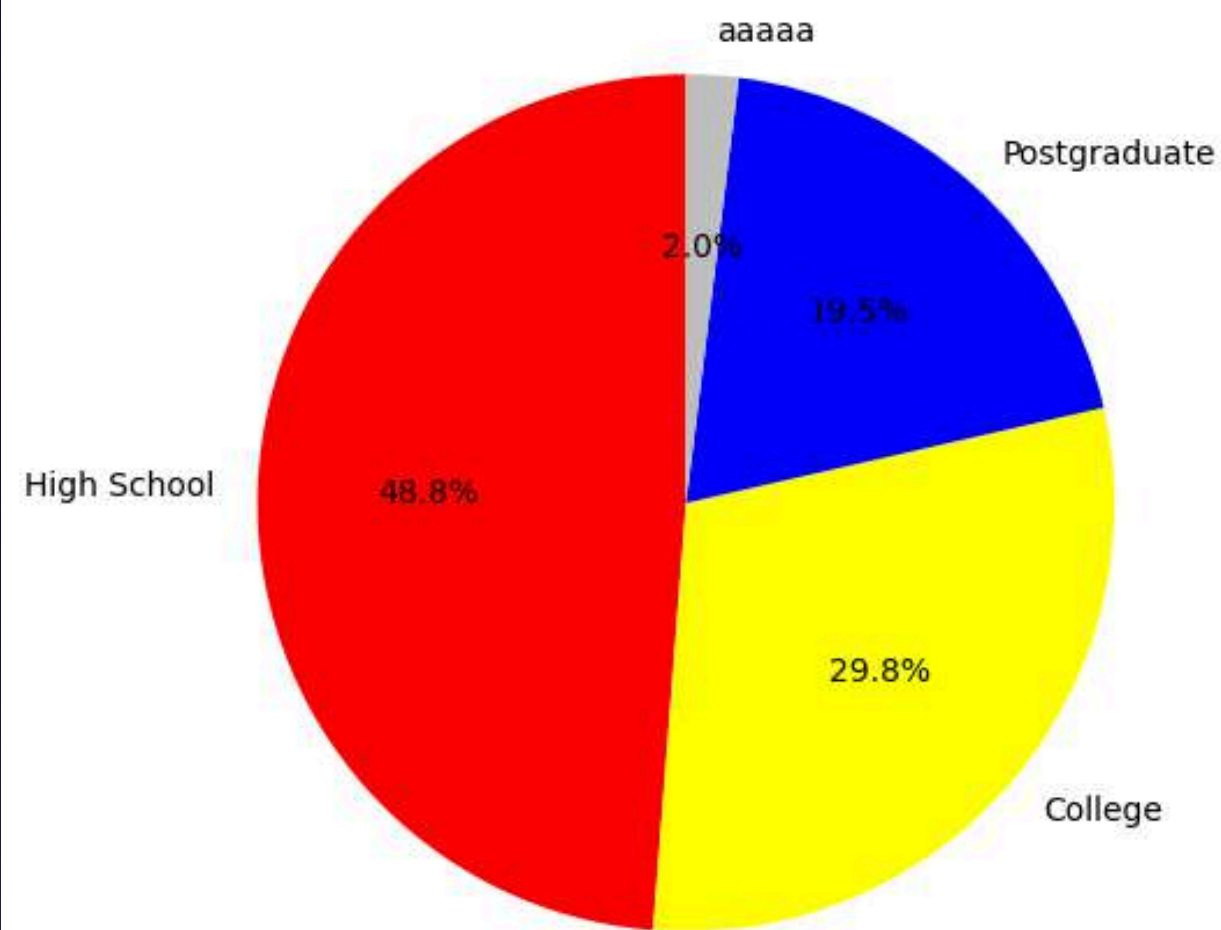


REVISION DE DATOS CATEGORICOS

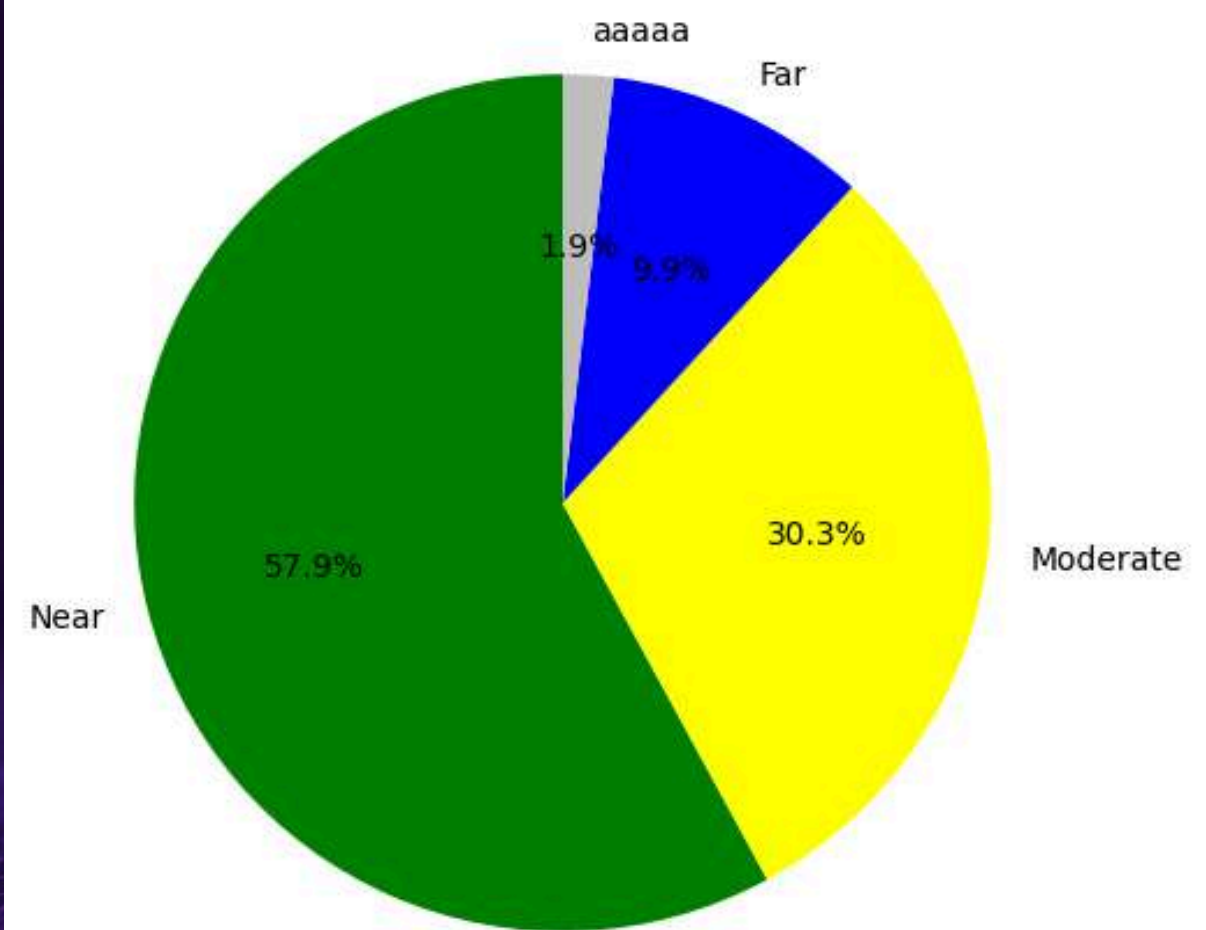


REVISION DE DATOS CATEGORICOS

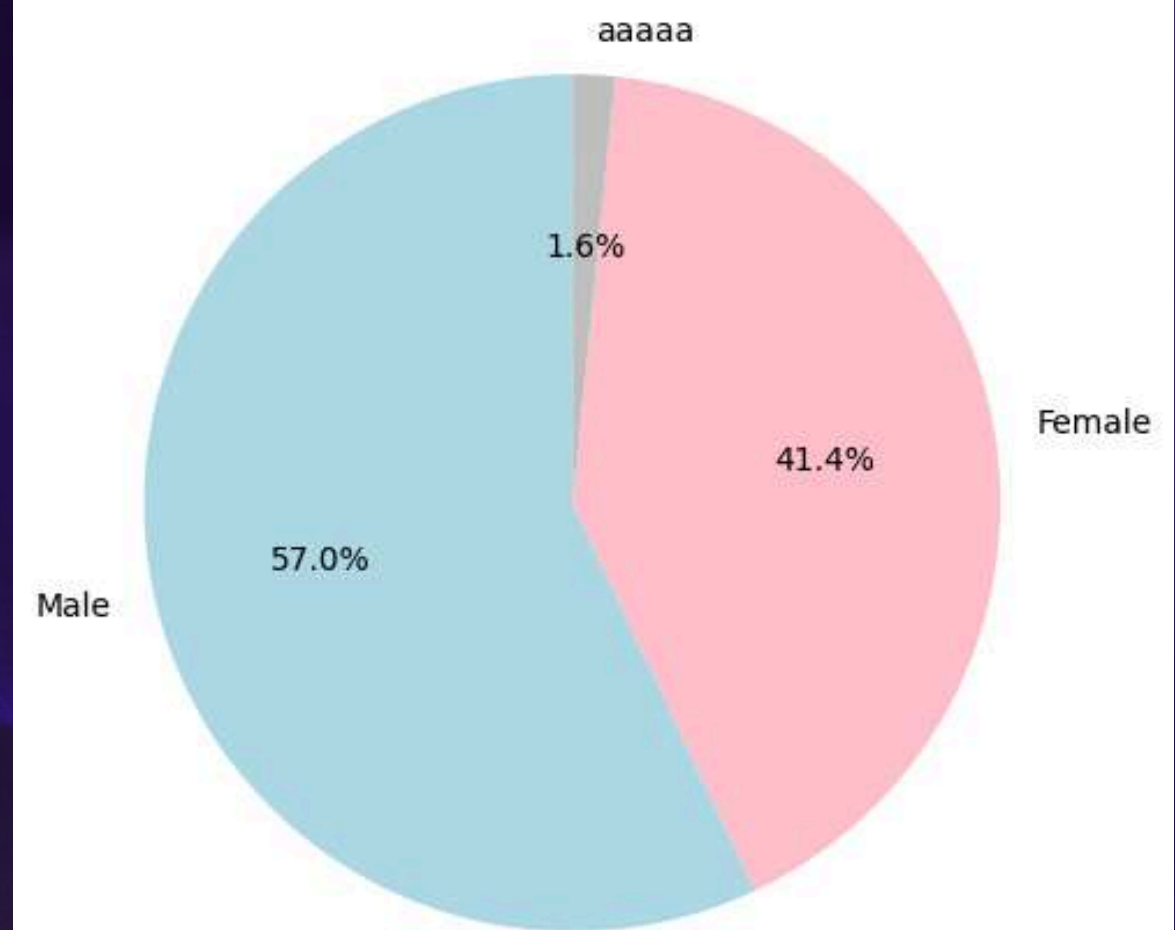
Distribucion por Nivel Educativo de los Padres



Distribucion por Distancia a Casa

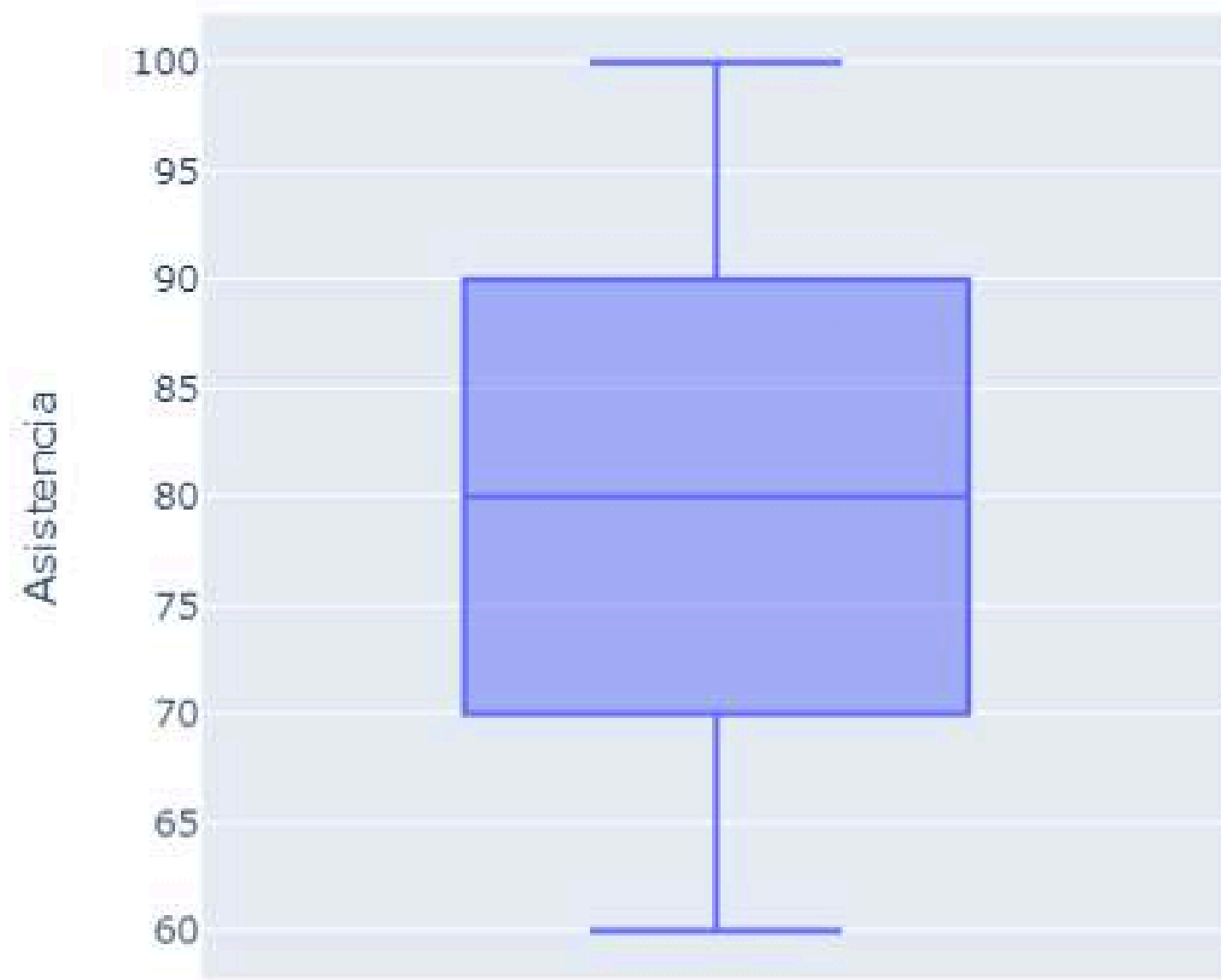


Distribucion por Genero

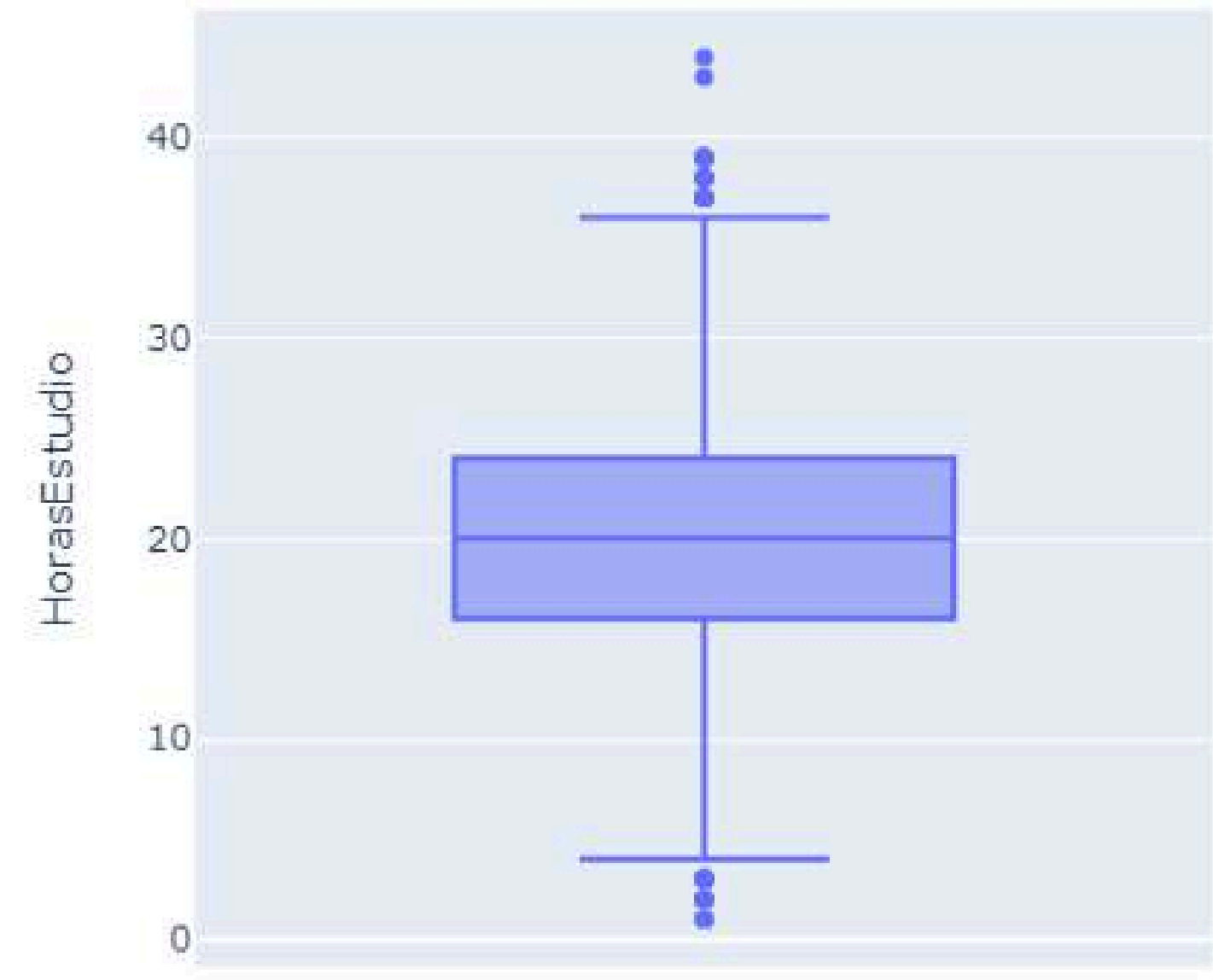


IDENTIFICACION DE DATOS ATIPICOS

Datos atípicos Asistencia



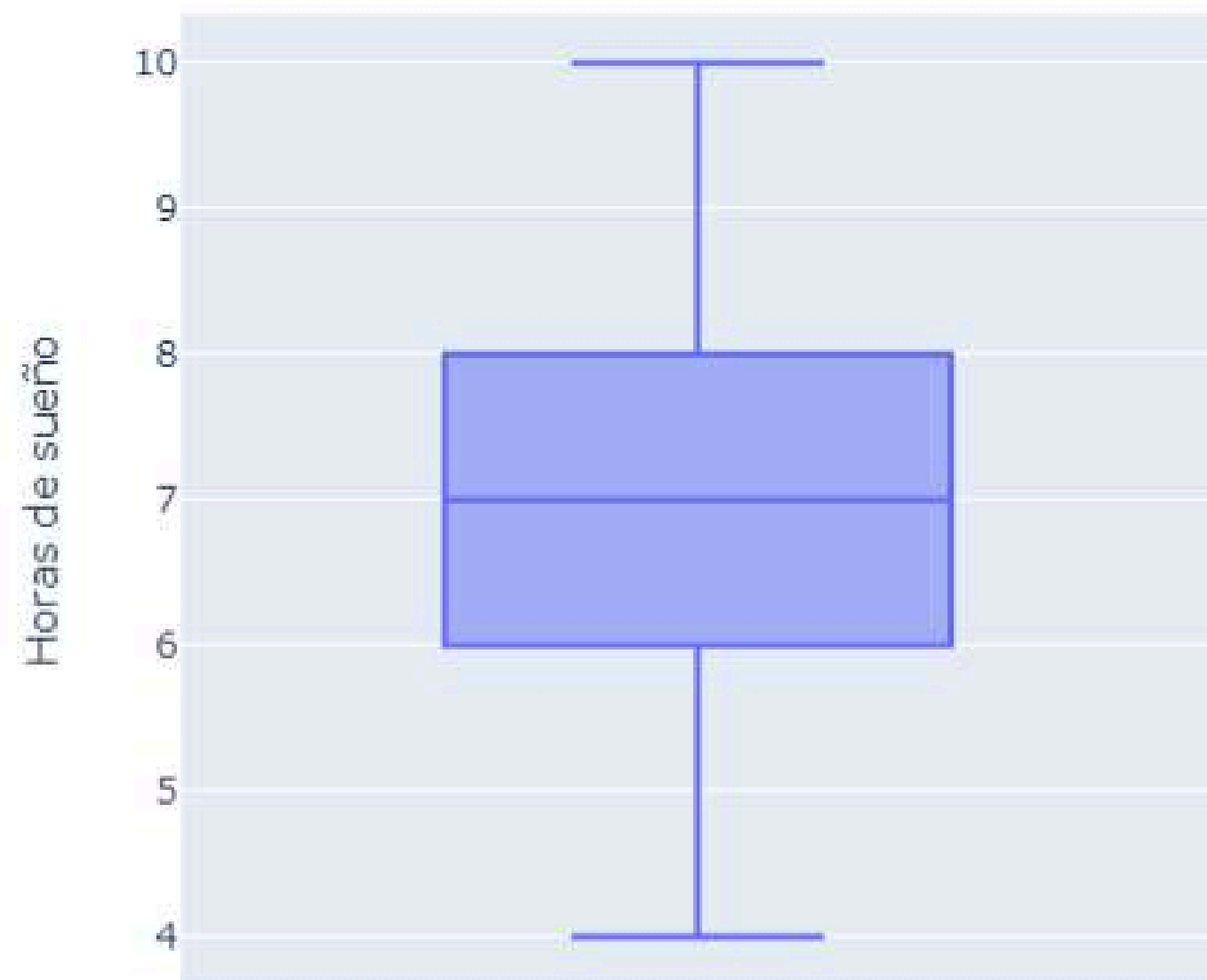
Datos atípicos Horas Estudio



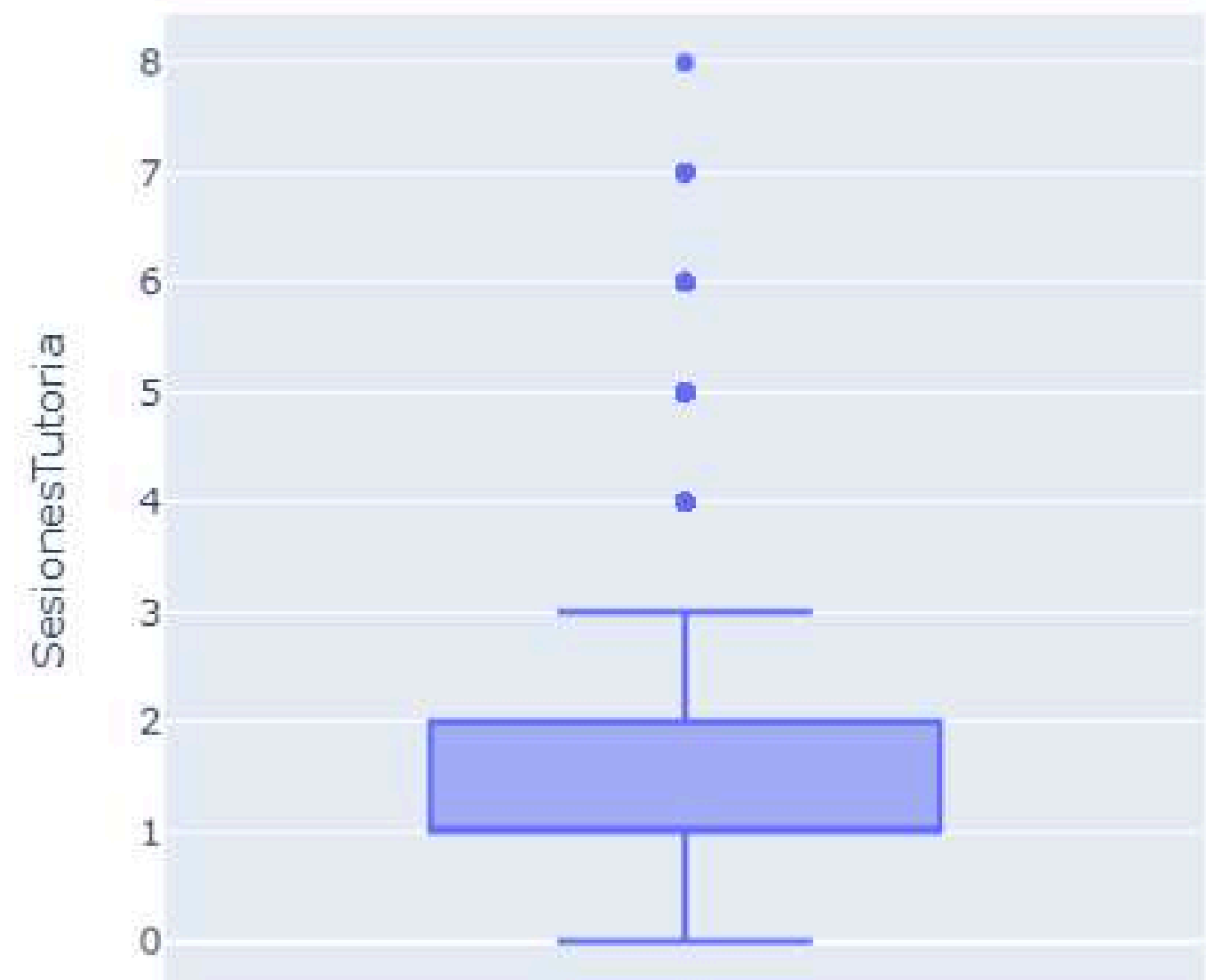
Contiene datos atipicos

IDENTIFICACION DE DATOS ATIPIICOS

Datos atípicos Horas Sueño



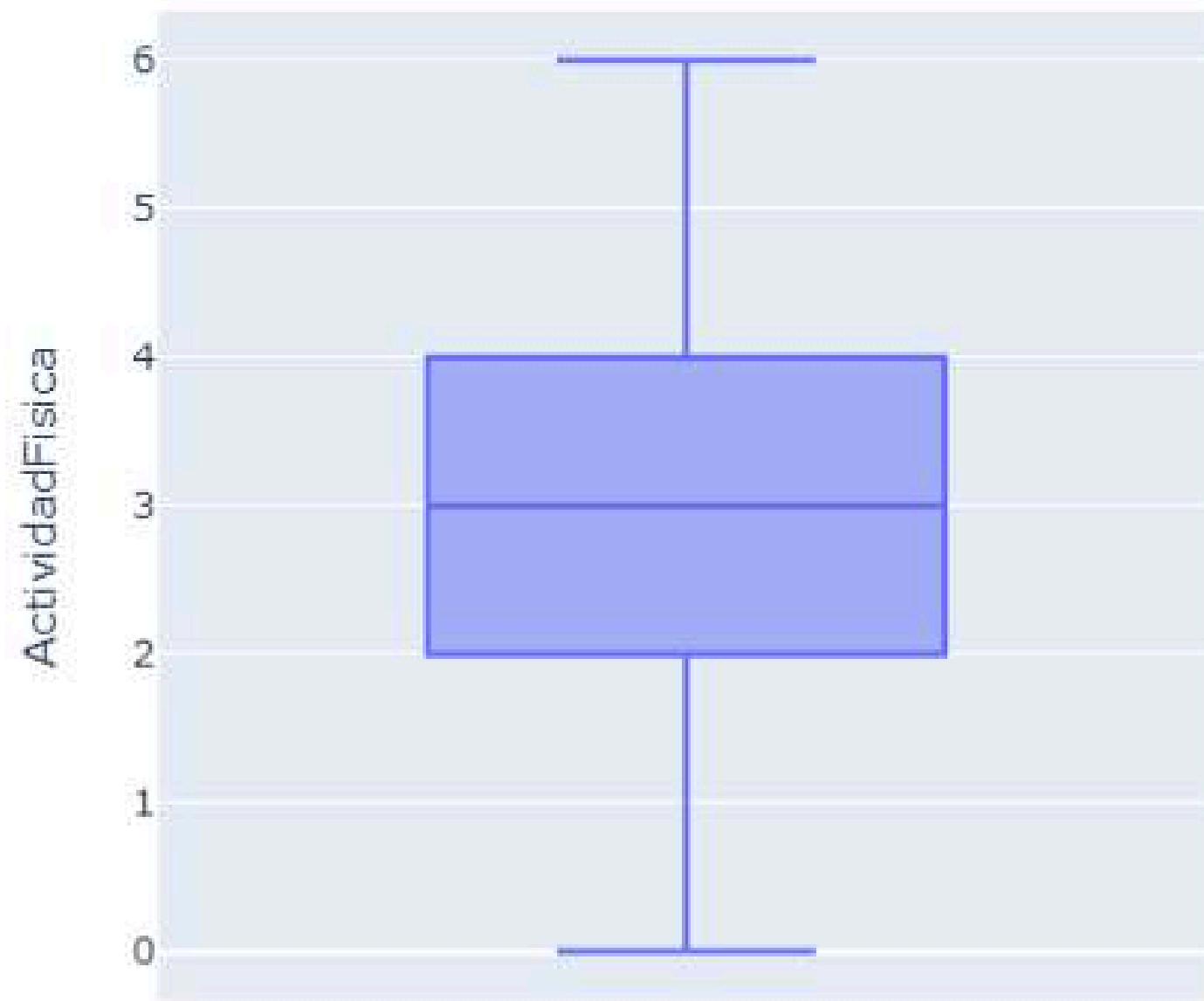
Datos atípicos Sesiones Tutoria



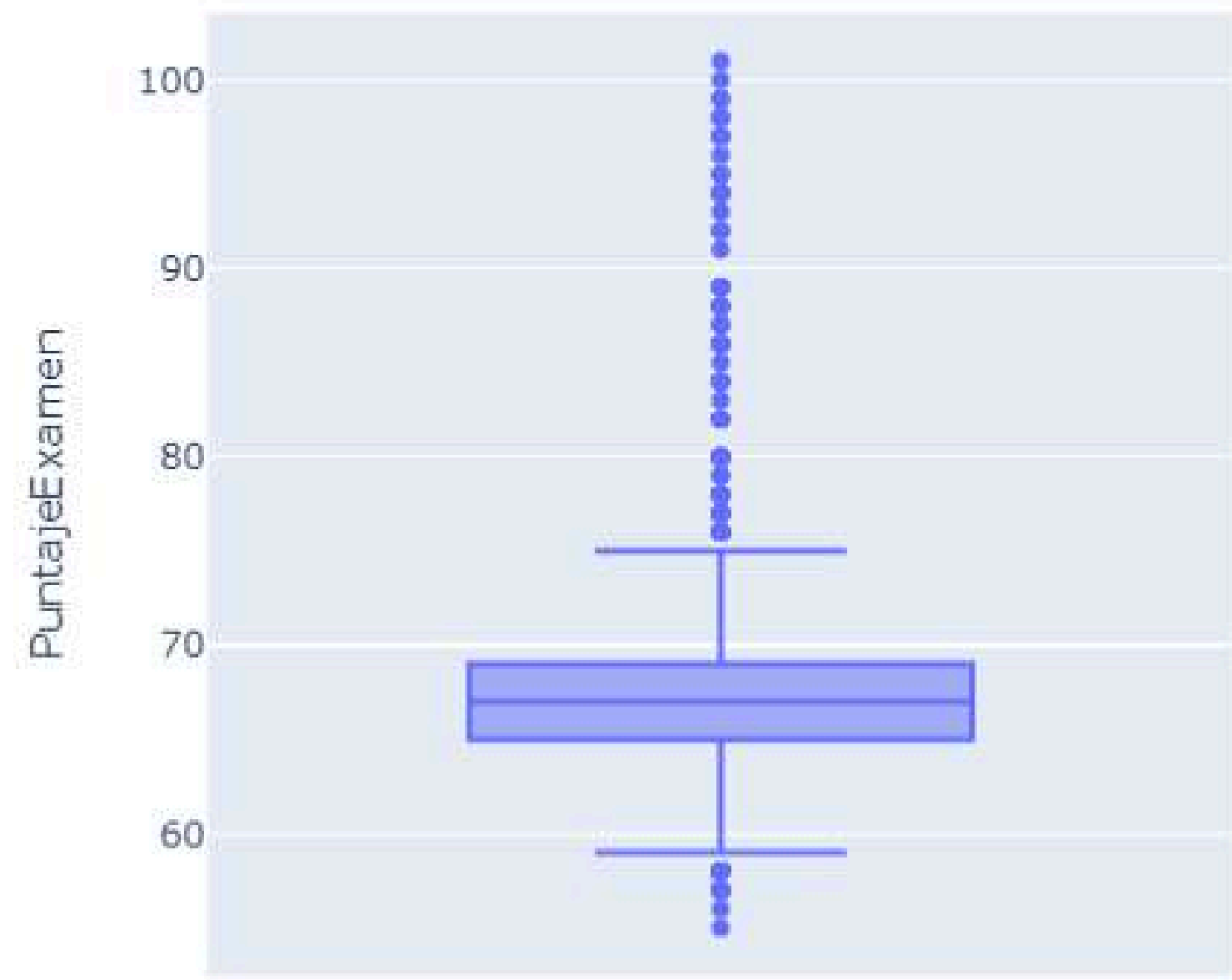
Contiene datos atipicos

IDENTIFICACION DE DATOS ATIPIICOS

Datos atípicos Actividad Fisica



Datos atípicos Puntaje Examen



Contiene datos atipicos

LIMPIEZA DE DATOS

Antes de comenzar a limpiar la base de datos, debe realizar un análisis preliminar para comprender la naturaleza y distribución de los errores



ELIMINACION DE DUPLICADOS

```
#Cantidad de duplicados  
df.duplicated().sum()
```

```
np.int64(54)
```

```
#Elimina los duplicados  
df2 = df.drop_duplicates()  
  
df2.duplicated().sum()
```

```
✓ 0.0s
```

```
np.int64(0)
```

ELIMINAR COLUMNA “MOTIVATION LEVEL”

```
#Eliminar columna en especifico
df2 = df2.drop(columns=['Motivation_Level'])
df2.info()
```

✓ 0.0s

<class 'pandas.core.frame.DataFrame'>

Index: 6699 entries, 0 to 6752

Data columns (total 19 columns):

#	Column	Non-Null Count	Dtype
0	HorasEstudio	6633 non-null	object
1	Asistencia	6649 non-null	object
2	InvolucramientoParental	6641 non-null	object
3	AccesoRecursos	6642 non-null	object
4	ActividadesExtracurriculares	6626 non-null	object
5	Horas de sueño	6642 non-null	object
6	ResultadosPrevios	6633 non-null	object
7	AccesoInternet	6646 non-null	object
8	SesionesTutoria	6632 non-null	object
9	IngresoFamiliar	6637 non-null	object
10	CalidadMaestro	6557 non-null	object
11	TipoEscuela	6642 non-null	object
12	InfluenciaCompañeros	6633 non-null	object
13	ActividadFisica	6631 non-null	object
14	ProblemasAprendizaje	6630 non-null	object
15	NivelEducacionParental	6550 non-null	object
16	DistanciaACasa	6580 non-null	object
17	Genero	6632 non-null	object
18	PuntajeExamen	6621 non-null	object

dtypes: object(19)

memory usage: 1.0+ MB

Eliminacion debido que no contiene
datos es decir todos son 'nan'

TRADUCCION DE DATOS CATEGORICOS

```
#Defino la traduccion
TradNivel = {
    'Low': 'Bajo',
    'Medium': 'Medio',
    'High': 'Alto'
}

TradGenero = {
    'Male': 'Hombre',
    'Female': 'Mujer',
}

TradEscuela = {
    'Public': 'Publica',
    'Private': 'Privada'
}

TradEducacion = {
    'High School': 'Bajo',
    'College': 'Medio',
    'Postgraduate': 'Alto'
}

TradDistancia = {
    'Near': 'Cerca',
    'Moderate': 'Medio',
    'Far': 'Lejos'
}

TradSiNo = {
    'Yes': 'Si',
}

TradInfluencia = {
    'Positive': 'Positiva',
    'Negative': 'Negativa'
}
```

✓ 0.0s

```
#Reemplazar los nombres de los pais en la columna 'Involucramiento Parental'
df2['InvolucramientoParental'] = df2['InvolucramientoParental'].replace(TradNivel)

#Reemplazar los nombres de los pais en la columna 'Involucramiento Parental'
df2['AccesoRecursos'] = df2['InvolucramientoParental'].replace(TradNivel)

#Reemplazar los nombres de los pais en la columna 'Actividades extracurriculares'
df2['ActividadesExtracurriculares'] = df2['ActividadesExtracurriculares'].replace(TradSiNo)

#Reemplazar los nombres de los pais en la columna 'Involucramiento Parental'
df2['AccesoInternet'] = df2['AccesoInternet'].replace(TradSiNo)

#Reemplazar los nombres de los pais en la columna 'Ingreso Familiar'
df2['IngresoFamiliar'] = df2['IngresoFamiliar'].replace(TradNivel)

#Reemplazar los nombres de los pais en la columna 'Ingreso Familiar'
df2['CalidadMaestro'] = df2['CalidadMaestro'].replace(TradNivel)

#Reemplazar los nombres de los pais en la columna 'Tipo Escuela'
df2['TipoEscuela'] = df2['TipoEscuela'].replace(TradEscuela)

#Reemplazar los nombres de los pais en la columna 'Tipo Escuela'
df2['InfluenciaCompañeros'] = df2['InfluenciaCompañeros'].replace(TradInfluencia)

#Reemplazar los nombres de los pais en la columna 'Problemas Aprendizaje'
df2['ProblemasAprendizaje'] = df2['ProblemasAprendizaje'].replace(TradSiNo)

#Reemplazar los nombres de los pais en la columna 'Nivel Educacion Parental'
df2['NivelEducacionParental'] = df2['NivelEducacionParental'].replace(TradEducacion)

#Reemplazar los nombres de los pais en la columna 'Distancia A Casa'
df2['DistanciaACasa'] = df2['DistanciaACasa'].replace(TradDistancia)

#Reemplazar los nombres de los pais en la columna 'Distancia A Casa'
df2['Genero'] = df2['Genero'].replace(TradGenero)
```

TRANSFORMACION A DATOS NUMERICOS

```
df2['HorasEstudio'] = pd.to_numeric(df2['HorasEstudio'], errors='coerce')
df2['Asistencia'] = pd.to_numeric(df2['Asistencia'], errors='coerce')

df2['Horas de sueño'] = pd.to_numeric(df2['Horas de sueño'], errors='coerce')
df2['SesionesTutoria'] = pd.to_numeric(df2['SesionesTutoria'], errors='coerce')

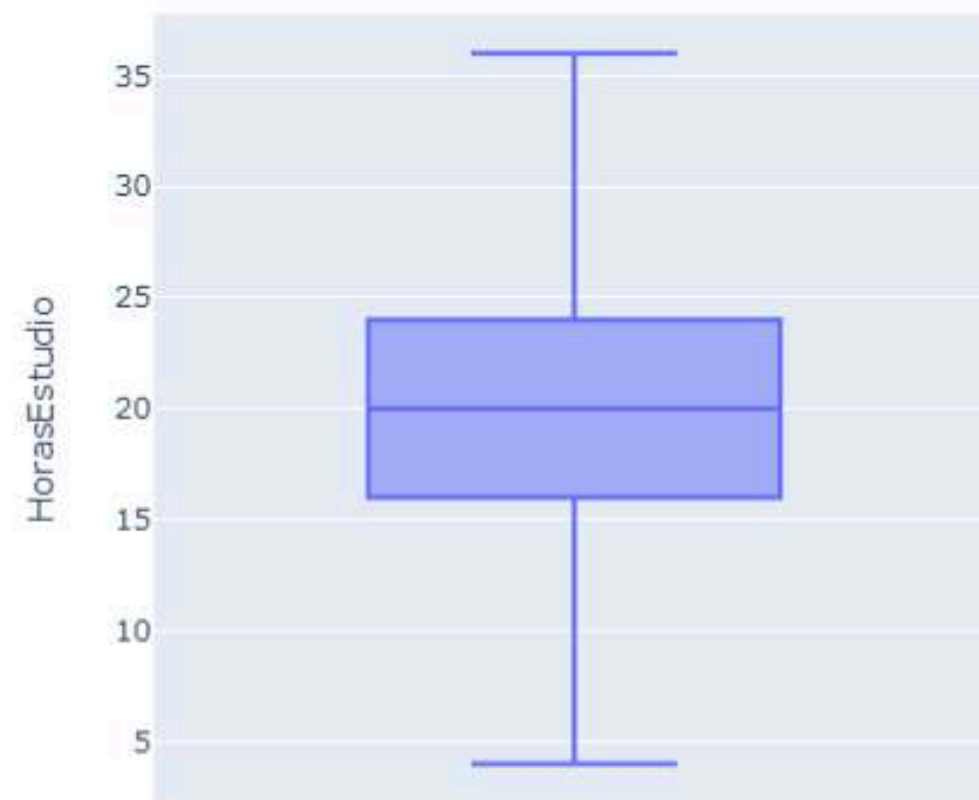
df2['ActividadFisica'] = pd.to_numeric(df2['ActividadFisica'], errors='coerce')
df2['PuntajeExamen'] = pd.to_numeric(df2['PuntajeExamen'], errors='coerce')
```

1

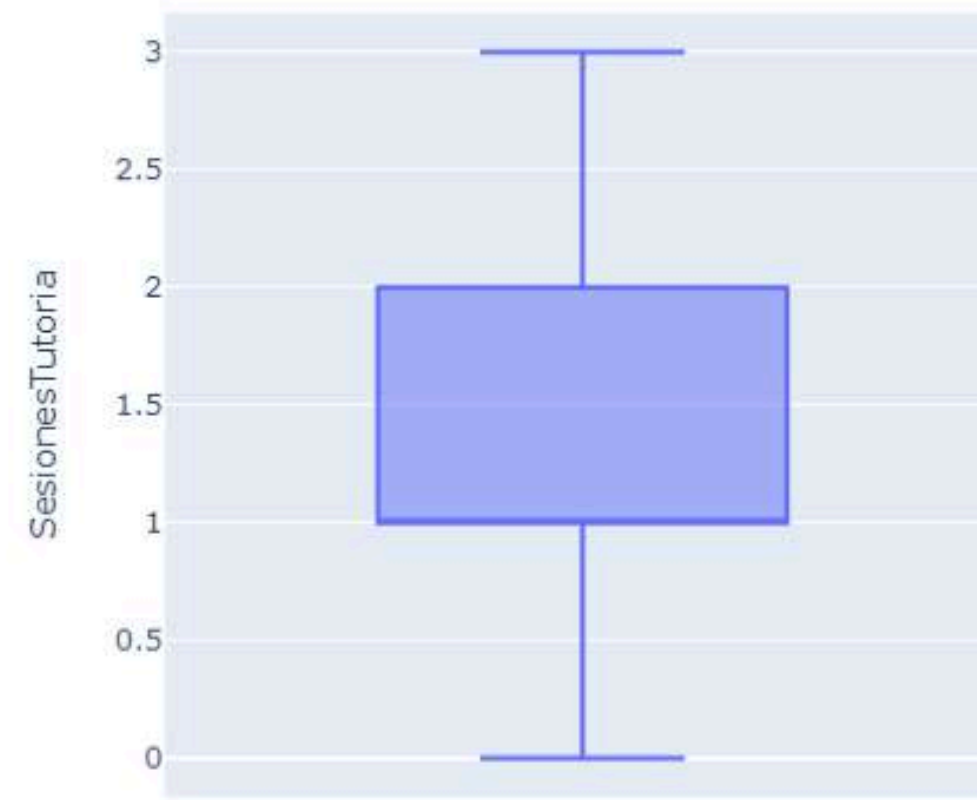
✓ 0.0s

DEPURACION DE DATOS ATÍPICOS

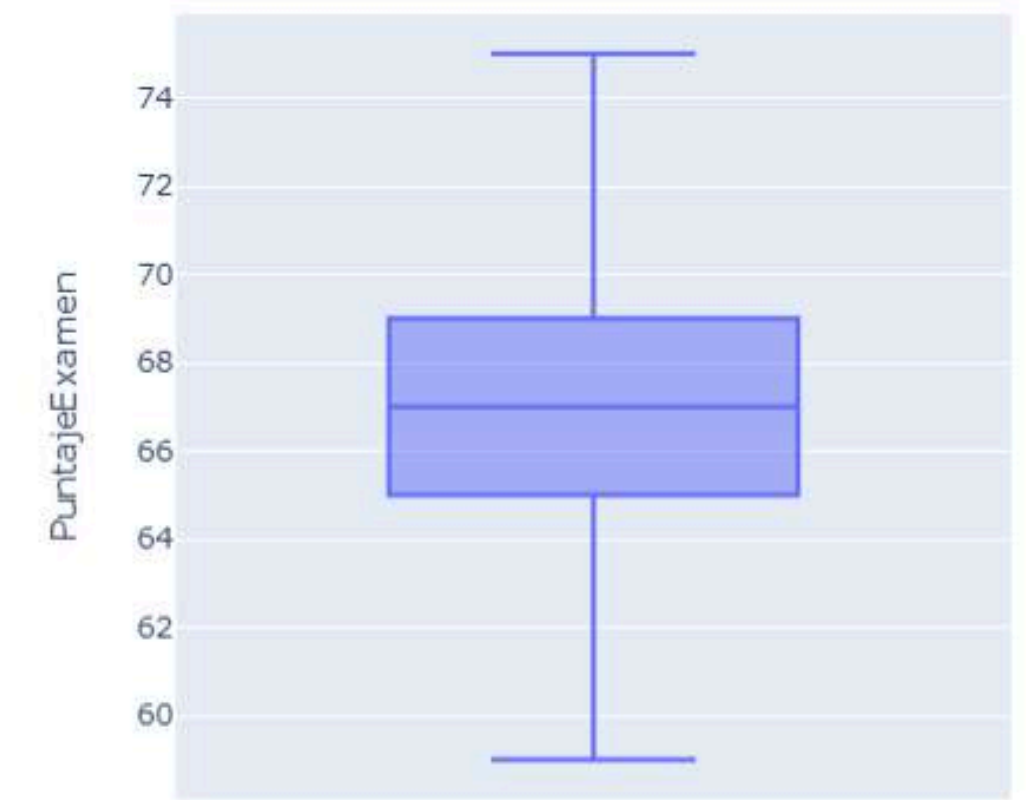
Datos atípicos Horas Estudio



Datos atípicos Tutoria



Datos atípicos eliminados Puntaje Examen



PROMEDIO DE COLUMNAS SIN DATOS ATIPICOS

```
PromAsistencia = round(df3_Asistencia['Asistencia'].mean())
```

✓ 0.0s

```
PromHorasEstudio = round(df4_HorasEstudio['HorasEstudio'].mean())
```

✓ 0.0s

```
PromHorasSueño = round(df3_HoraSueño['Horas de sueño'].mean())
```

✓ 0.0s

```
PromTutorias = round(df4_SesionesTutoria['Horas de sueño'].mean())
```

✓ 0.0s

```
PromActividadFisica = round(df3_ActividadFisica['Horas de sueño'].mean())
```

✓ 0.0s

```
PromPuntaje = round(df4_PuntajeExamen['PuntajeExamen'].mean())
```

✓ 0.0s

RELLENAR VALORES NAN CON PROMEDIO

```
df2['HorasEstudio'].fillna(PromHorasEstudio, inplace=True)
df2['Asistencia'].fillna(PromAsistencia, inplace=True)
df2['Horas de sueño'].fillna(PromHorasSueño, inplace=True)
df2['SesionesTutoria'].fillna(PromTutorias, inplace=True)
df2['ActividadFisica'].fillna(PromActividadFisica, inplace=True)
df2['PuntajeExamen'].fillna(PromPuntaje, inplace=True)
df2['ResultadosPrevios'].fillna(PromPuntaje, inplace=True)
```


ELIMINAR VALOR "AAAAAA"

```
df3['InvolucramientoParental'].replace("aaaaa",np.nan , inplace=True)
df3['AccesoRecursos'].replace("aaaaa",np.nan , inplace=True)
df3['ActividadesExtracurriculares'].replace("aaaaa",np.nan , inplace=True)
df3['AccesoInternet'].replace("aaaaa",np.nan , inplace=True)
df3['IngresoFamiliar'].replace("aaaaa",np.nan , inplace=True)
df3['CalidadMaestro'].replace("aaaaa",np.nan , inplace=True)
df3['TipoEscuela'].replace("aaaaa",np.nan , inplace=True)
df3['InfluenciaCompañeros'].replace("aaaaa",np.nan , inplace=True)
df3['ProblemasAprendizaje'].replace("aaaaa",np.nan , inplace=True)
df3['NivelEducacionParental'].replace("aaaaa",np.nan , inplace=True)
df3['DistanciaACasa'].replace("aaaaa",np.nan , inplace=True)
df3['Genero'].replace("aaaaa",np.nan , inplace=True)
```

RELLENA DATOS NAN CON "NODATA"

```
df3['InvolucramientoParental'].fillna("NoData", inplace=True)
df3['AccesoRecursos'].fillna("NoData", inplace=True)
df3['ActividadesExtracurriculares'].fillna("NoData", inplace=True)
df3['AccesoInternet'].fillna("NoData", inplace=True)
df3['IngresoFamiliar'].fillna("NoData", inplace=True)
df3['CalidadMaestro'].fillna("NoData", inplace=True)
df3['InfluenciaCompañeros'].fillna("NoData", inplace=True)
df3['ProblemasAprendizaje'].fillna("NoData", inplace=True)
df3['NivelEducacionParental'].fillna("NoData", inplace=True)
df3['DistanciaACasa'].fillna("NoData", inplace=True)
df3['Genero'].fillna("NoData", inplace=True)
```

REVISION FINAL

```
df3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
Index: 6600 entries, 0 to 6752
```

```
Data columns (total 19 columns):
```

#	Column	Non-Null Count	Dtype
0	HorasEstudio	6600 non-null	float64
1	Asistencia	6600 non-null	float64
2	InvolucramientoParental	6600 non-null	object
3	AccesoRecursos	6600 non-null	object
4	ActividadesExtracurriculares	6600 non-null	object
5	Horas de sueño	6600 non-null	float64
6	ResultadosPrevios	6600 non-null	float64
7	AccesoInternet	6600 non-null	object
8	SesionesTutoria	6600 non-null	float64
9	IngresoFamiliar	6600 non-null	object
10	CalidadMaestro	6600 non-null	object
11	TipoEscuela	6444 non-null	object
12	InfluenciaCompañeros	6600 non-null	object
13	ActividadFisica	6600 non-null	float64
14	ProblemasAprendizaje	6600 non-null	object
15	NivelEducacionParental	6600 non-null	object
16	DistanciaACasa	6600 non-null	object
17	Genero	6600 non-null	object
18	PuntajeExamen	6600 non-null	float64

```
dtypes: float64(7), object(12)
```

```
memory usage: 1.0+ MB
```

Tipos de datos correctos

Sin datos invalidos

Sin valores NaN

Sin Duplicados



```
df3.duplicated().sum()
```

```
[85]
```

```
✓ 0.0s
```

```
...
```

```
np.int64(0)
```


The background is a deep purple color. A faint, glowing wireframe grid is visible, creating a sense of depth and movement. In the top-left and bottom-right corners, there are stylized, 3D-rendered brains in a lighter shade of purple. The text is centered in the middle of the image.

**¡GRACIAS POR
LA ATENCIÓN!**